

In Japan, nobody makes a living writing game programs. In fact, it would seem that the Japanese consider the title "professional game programmer" a contradiction in terms.

Although writing software for business applications is thought fairly respectable, game programming is held in much lower esteem. It has what the Japanese describe as a "negative aspect" to it. The Japanese view game programming as trivial—kind of like a game itself—not professional, and perhaps not even quite honorable. Certainly, becoming a game programmer is not an appropriate goal for a serious-minded, creative young person in Japan to aspire to.

Another View. Minoru Nakazawa, president of Star Craft, a Japanese software exporting firm, disagrees with this assessment. He wants the situation in his country—both the prevailing attitude toward game programming and game programmers' feelings about themselves—to change. Looking forward to a time when Japanese game programmers work in a positive environment and command the admiration and respect of their countrymen, Nakazawa is doing all he can to put game programming in a new perspective.

A significant step in this direction took place in August. Accompanied by Gary Carlston, of Broderbund

Software, Bill Budge, author of *Raster Blaster* and other popular game programs, visited Japan at Nakazawa's invitation. The purpose of Budge's visit was to create an opportunity for dialogue between Japanese game programmers and the American; the setting was an informal gathering to be attended by Nakazawa, Budge, and several Japanese programmers, including Tony Suzuki, author of *Alien Rain*, Jun Wada and Ken Iba, coauthors of *Snoggle*, and Raly Yamaguchi, a talented business programmer. It was Nakazawa's hope that such a meeting would both educate and encourage Japanese game programmers.

With a translator facilitating the discussion, the day's topics ranged from hardware to software, from game programming to business programming, from the conditions of the present to the challenges of the future. And, after the tentative first ten minutes, the language barrier was hardly noticeable, so eager were the participants to ask questions, share experiences, and discuss concerns.

Hardware Breaks the Ice. After welcoming Budge and the other programmers and expressing his hopes for the day, Nakazawa asked Budge to comment on how Japanese hardware is assessed in the United States. Budge assured his

Gamemakers' Summit

Bill Budge Chats with Star Galaxy



listeners that Japanese technology is well thought of here, so well thought of, in fact, that many people believe the Japanese will eventually take over the American market. As a result, Budge explained, there are almost no hardware companies left. Instead, the response of American companies has been to become software companies. "If a company builds a new computer, ten people work on the hardware and ninety people work on the software."

By contrast, the Japanese market for software is a limited one. Japanese consumers, unlike their counterparts in this country, have not yet begun purchasing microcomputers in great numbers for personal/home use. In addition, so many varieties of microcomputer are produced in Japan that no one, two, or three in particular have emerged as favorites for which the majority of software is designed. Not much game software is made in Japan and most of what is produced there finds its market in the United States; hence, the existence of an export company like Nakazawa's Star Craft.

A major difference between the Japanese and the American game programming climates became apparent when Budge talked about how strange it is to walk into a computer store and see his programs on shelf or to overhear people

talking about them. "If any of you go to America, it's really fun to go to a store and see your program and just wait for somebody to come and look at it or buy it." The Japanese programmers could imagine the experience Budge described, but they could not identify with it firsthand. Their own programs are not very likely to be on the shelves of computer stores in Japan.

Instead, both Suzuki and Wada spoke of the lack of confidence they tend to feel in their abilities to make their livings as game programmers. They contrast their perception of the American attitude of confidence with the more tentative attitude they experience in Japan—an unstated but powerful feeling of uncertainty that begins to convince them that they can't do it, that they probably won't succeed, so it's rather pointless to try.

Confidence Takes Time. Budge believes that it is only a matter of time before Japanese consumers catch microcomputer fever, and that once they do, an installed user base will come into being. When this happens, the Japanese demand for software of all sorts will increase dramatically, ensuring programmers of a market for the software they produce. And it's likely that Japanese programmers will be inspired to create more and better software if they feel confident that their efforts will be supported in their own country.

But for right now, Japanese game programmers cannot be sure of such respect and appreciation in Japan. When Budge asked Suzuki and Wada whether they planned to make professional game programming their life's work, the uncertainty about the future prospects for professional game programmers in Japan surfaced again. As much as he enjoys game programming, when it comes to a career, Suzuki thinks about becoming a scientist of some sort, a chemist maybe, instead of a game programmer.

Budge's heartfelt response to Suzuki: "I hope you'll change your mind. You're a very good programmer. In the future game programs will become much harder to write, and I think writing them will be very respectable. That's a challenge to us, because the money that will be there won't be free; you'll have to be able to write the most popular program ten years from now, which none of us may be able to do."

As Budge sees it, the possibilities of game programming and graphics are vast and have just barely been explored. "Very few people," asserts Budge, "are making an effort to extend the idea of a personal home computer game or making the effort to do more sophisticated graphics. The field of computer graphics is very big, and game programmers have used but a small corner of it. . . . I am thinking of personal computer games as being more than what they are now, which mainly involves shooting things that move. Personal computer games would be very different. I think we're just beginning to explore the possibilities."

Games Plus Business Equals Benefits Plus Compromises. According to Nakazawa, Japanese microcomputer manufacturers, like those in the United States, are attempting to include "the best of everything: games and business" in a personal computer. The Japanese realize that such an approach requires design compromises.

Yet the interrelationship developing between business programs and game programs has positive consequences for

Pictured below: Bill Budge, Jun Wada, and Tony Suzuki.

Craft's of Programmers

by
**Jean
Varven**



both fields. People's concerns about business applications have resulted in the creation and use of new hardware that cannot help but benefit game programming. And business programs are also changing; graphics are becoming an integral part of business applications programs.

Budge has tried his hand at creating both serious programs and game programs. He enjoys business programming—in fact, he's working right now on developing a new word processor—but says, "I think I find game software more challenging and more fun. It allows you to be more creative and to try more sophisticated techniques; if things don't work, it's not a disaster."

New CPUs Welcome—with Some Reservations. Budge wants to see the development and widespread use of better, more powerful CPUs. Both the graphics involved in game programming and the speed at which game programs operate push hardware to its limits. "A business program—you write it, and it runs as fast as it runs. But a game program, you write it, and, if it's not fast enough, you throw it away."

Wada, Suzuki, and Budge all agreed that that's a frustrating experience to come up against the limits of the smaller CPUs. They have the feeling at times that they have simply run out of things to do. This contributes, in Budge's view, to the lack of understanding that some people have about game programming.

Budge sees learning about the capabilities and construction of new microprocessors as one of the essential challenges to game programmers in the future. "The reason I want a faster CPU is because really advanced computer graphics use many data structures. Very complicated techniques like curved surfaces and shading require complicated data structures." A microprocessor like the 6502 that drives the Apple II and the Atari lacks the power that makes games all they can be.

The Japanese programmers expressed some uncertainty at the thought of learning the ins and outs of a whole new microprocessor. "Wouldn't you rather wait until the right computer's on the market so you wouldn't have to keep restudying it?" joked Wada.

While Budge could relate to these feelings, he encouraged a positive attitude. Learning a new microprocessor soon after its introduction gives a programmer certain advantages. He is in on the ground floor of a new market and has plenty of time to develop his expertise and to explore the possibilities.

A Special Blend. Of course, the programmers agreed, there can also be a disadvantage in being too early. When there's not yet a base of computer owners to buy new software programs, some of the people who create programs early on may be passed by. And they also had to agree with Nakazawa's point that Suzuki is a real exception to all the rules; although he started late on the Apple, he surely caught up in a hurry.

And in fact each one of the programmers who met together in Japan this past August is exceptional. As Nakazawa pointed out, "Among the Apple users in Japan, there are quite a few people who are very knowledgeable and who have the ability to write good programs; on the other hand, there are very few among them who, like Tony

Suzuki and Jun Wada, complete really good quality, marketable programs."

Budge concurred and in the process of responding, went on to enumerate some of the attributes of a successful game programmer. "I think it's the same in the United States as it is in Japan. Most people don't have the combination of skills—it's kind of specialized. To write a good game, you have to know assembly language, which rules out quite a few people; you have to like to play with graphics, which rules out more people; and in the end, there aren't that many people left who can write games. And the number who can finish a program is another small fraction."

Budge suggests that people are likely to have a deeper understanding of what's involved in graphic game programming if they try some programming themselves. "People send me letters, and basically, the question is: What's the secret of hi-res programming? Not specific questions. They want to know the secret. . . . They see a ball bounce off something and they think, well, that's the way the computer does things; but that's not at all true. It's a great deal of work to make things the way you want them."

Users groups or clubs, common in both America and Japan, can foster a better, more informed understanding of the hard work and creative energy that go into programming. Despite the piracy problems in both countries, Budge and all the others present agreed that clubs are, for the most part, a positive force. They look forward to a time when the users who spend time and effort breaking the codes of existing software programs begin instead to develop their skills and use them to create programs of their own.

Moving into the Future. Game programming continues to evolve. Recently, we've seen software become a consumer product in which program, documentation, packaging, and advertising all work together. As for the future, Budge sees more complex graphics and predicts a shift in the emphasis of personal computer games—away from speedy home-arcade games and toward games with more diverse entertainment value.

Budge also envisions more collaboration in the creation of game programs. He suggests that computer game companies of the future may be modeled after the sort of technical group that Star Wars producer George Lucas has put together to produce computer graphics for films, with computer graphics experts working in teams with systems programmers.

When individuals who want to understand and learn from one another decide to collaborate on a project, some marvelous things can happen. This sharing of knowledge and ideas between people who respect one another's skills may give each individual the opportunity to express himself in a new way.

Finally, deciding whether to make game programming one's profession is not a simple matter. It's an individual decision in which many factors play a part. But whatever career choices they make later on, the programmers who met in Japan this past August know the meaning of professionalism. It has to do with respect, and being able to respect themselves for the integrity of their own programs is what matters most. 51